Description

Insert A'>

METHOD AND SYSTEM FOR SHIPPING/MAILING

Technical Field

The invention relates to shipping/mailing techniques, more particularly utilizing distributive
5   computerized technology.

Background of the Invention

Many offices/organizations process large numbers of mail pieces or parcels and utilize different shipping or mailing carriers such as the United States
10   Postal Service (USPS), United Parcel Service (UPS), Federal Express (FedEx), RPS and DHL, for example. For each mail piece, the carriers require shipping/mailing information including the delivery address and, typically, further instructions such as the class of
15   service, for example.

The required information may be supplied by manual entry, e.g. using the carrier's proprietary software. Such entry tends to be inefficient and error prone.

20

Summary of the Invention

The present invention aims at more efficient and error-free processing of shipping/mailing information. Measures are taken for reducing manual work
25   and validating the information, including utilization of optical scanning, character recognition (OCR) and bar codes, and reference to standard address databases in a distributive-processing technique, e.g. client-server or peer-to-peer.

In addition to the addressee, a user of the technique may specify the carrier and/or a class of service to be used for delivery. Alternatively, choice of a delivery option can be provided automatically, based
5   on predefined rules. At a user site, delivery information may be entered by typing, by importing from a personal or public database/list, or by scanning by an optical character recognizer (OCR), for example.

An entered delivery address can be checked
10 against the USPS Address Matching System (AMS) database to verify its validity. If the address fails to check out, possible valid addresses can be offered automatically for the user's consideration. Automatically also, addresses can be standardized, e.g.
15 as to font and format, and for readability. Additional data may be appended, e.g. an internal billing code and/or a tracking ID.

Shipping/mailing data as provided or generated can be printed onto a label or other suitable medium,
20 readable to a human and/or in an encoded form, e.g. a 2-dimensional bar code as based on a 2-D symbol standard such as PDF-417 or Data Matrix, for example. With the label affixed, e.g. detachably, a parcel or mail piece is ready for forwarding to a shipping/mailing room/location.
25   Preferably, with the label including a bar code, shipping/mailing information can be scanned for automated processing at the shipping location, to print the selected shipper's actual shipping label and postage if required. To facilitate tracking, the
30 shipping/mailing information may be uploaded to the shipper, e.g. to UPS Online.

Brief Description of the Drawing

Fig. 1 is a diagram illustrating mail piece
35 origination.

Fig. 2 is a diagram illustrating mail piece processing at a shipping/mailing center.

Fig. 3 is an example of a computer opening/main screen view in mail piece origination.

5   Fig. 4 is a diagram of a distributive network as can be used in mail piece origination, including address validation and standardization.

Fig. 5 is a schematic for shipping/mailing address validation and standardization.

10   Fig. 6 is a state diagram for automated shipping/mailing address standardization.

Fig. 7 is a data flow diagram for automated address printing.

Fig. 8 is a state diagram for automated label
15 generation.

Fig. 9 is a state diagram for automated address database importing.

Fig. 10 is a state diagram for automated address standardization and validation.

20   Figs. 10 and 11 are state diagrams for revenue protection.

Fig. 12 is a state diagram for automated feature authorization.

Fig. 13 is a state diagram for automated
25 safeguarding against unauthorized access to address standardization/validation.

Fig. 14 is a state diagrams for automated license registration.

Fig. 15 is a state diagram for automated seat
30 feature enforcement.

Detailed Description

Features as described herein with reference to the drawing have been implemented in an exemplary system
35 here designated as Addressing and Bar Code (ABC)

Link/Host.   The features are not required all to be
included in a single embodiment of the invention, but can
be used individually or in any suitable combination
within various preferred embodiments.   Conveniently in

5   implementation, a suitable programming language is used,
e.g. C++.

Figs. 1 and 2 illustrate over-all processing in
shipping/mailing, e.g. at a large office facility.
Specifically, Fig. 1 illustrates origination or

10   generation of mail pieces at an enterprise network 100,
and Fig. 2 their processing at a shipping/mailing center
103 where the mail pieces are further processed to
shipping carriers such as the Post Office, UPS, RPS,
FedEx and DHL, for example.

15   Fig. 1 shows a label 105 comprising a bar code
110, generated at a enterprise network site 100 for
processing a mail piece 115. A user at a terminal 101 of
the site 100 enters shipping information for the mail
piece 115, such as shipping destination, originator

20   identification, carrier, shipping class and declared
value of the contents.   The shipping information is
encrypted and included in the bar code 110 on the label
105.   The bar code 110 may be based on the PDF-417, Data
Matrix, or other 2D-symbol standard.   The label 105,

25   which includes the entered shipping information and the
bar code 110, is printed on a network or local printer of
the site 100, and placed on the mail piece 115 for
forwarding to the center 103 of Fig. 2 for
shipping/mailing.

30   Fig. 2 shows the bar code 110 for the mail
piece 115 being read using a bar code scanner 120
connected to a terminal 125 at the shipping/mailing
center. The terminal 125 has suitable bar code
recognition and decryption software for extraction and

35   decryption of the shipping information from the bar code

110. The terminal 125 converts the shipping information
into the appropriate format of the carrier selected for
the mail piece 115. The converted information is
uploaded to the shipping software of the selected
5   carrier, e.g. UPS Online, and the terminal 125 instructs
a thermal printer 130 to print a shipping label 135 for
use by the carrier. With the shipping label 135 affixed,
the mail piece 115 is ready for processing by the
selected carrier.

10   Fig. 3 shows a graphical user interface (GUI)
or screen display for processing at the terminal 101,
with shipping by the USPS being shown as an example. The
display resembles typical text processor screens,
including a row 151 of menu buttons, a row 152 of icons,
15   a shipping class display 153 as selected by one of the
click tabs 154, here for the USPS, an address text
display 155, a special services selection display 156, an
originating department information display 157, a
multiple-label button 158, a print button 159, an address
20   book access button 160, a "remove" button 161 and a
shipping directions button 162. Functions are actuated
and controlled by typing, and by familiar clicking on
buttons, tabs and icons.

            It has been recognized that the use of
25   conventional bar codes for the labels generated at
network 100 for processing at a shipping/mailing center
103 may be susceptible to fraudulent circumvention. For
example, a conventional bar code on label 105 might be
readable by an unauthorized, conventional bar code
30   reader. The use of unauthorized systems and components
may undermine the integrity and performance of the
shipping process.

            As a countermeasure, the shipping information
for the mail piece 115 is encrypted before it is used to
35   generate the bar code 110. The terminal 125 includes a

decryption algorithm for the data read by the bar code reader 120 from the bar code 110. Unauthorized systems, without the decryption algorithm will be unable to process the encrypted shipping information from the bar

5   code 110.

Further to deter the use of unauthorized equipment at shipping/mailing room center 103, the shipping address and information for a mail piece can be shuffled in accordance with a predetermined shuffling

10   algorithm prior to encryption. For example, the order of first and last names of a recipient may be reversed prior to encryption. At the mailroom terminal 125, a rearrangement algorithm will then undo the shuffling. Shuffling and rearrangement algorithms can be updated

15   periodically to prevent their discovery upon inspection of the shuffled shipping information.

While use of the scanner 120 eliminates the likelihood for error in transferring the shipping information onto the shipping label 135, without further

20   validation there remains a concern with error at the source, e.g. a user at the terminal 101 entering erroneous shipping information. A resulting invalid shipping address may remain undetected until the carrier fails to deliver the mail piece 115. This concern can be

25   minimized by measures as follows:

Fig. 4 shows a user network 100 for use with Windows NT, featuring address validation using a database provided by the USPS, with validation being facilitated by standardizing addresses as to their format. The USPS

30   address database service, known as its Address Matching System (AMS), includes on a CD-ROM all valid U.S. addresses in a standardized format. Updated versions are provided periodically under a license agreement.

The network 100 comprises a network server 200

35   and a network hub 205, providing network services to

client terminals 210, 215, 220, and 225. The network 100
may be a packet-switched network for transporting
information in accordance with the standard transmission
control protocol/internet protocol (TCP/IP). Remote
5   access is provided for terminal 225 by a dial-up/Internet
connection through the modem 230.

        Windows operating systems, e.g. Windows 95, 98
or NT are installed at the server 200 and terminals 210,
215, 220 and 225 for communicating amongst one another.
10 Further installed at the terminals 215, 220 and 225 is
software here designated as ABC Link and, at the terminal
210, software designated as ABC Host. The latter
includes an AMS capability for making use of the AMS CD
in a CD-ROM drive 235. A hardware key 240 is connected
15 at a communication port of the terminal 210, representing
a contractual safeguarding element.

        For the host terminal 210, use of Windows NT is
advantageous in that it provides a launch service that
keeps ABC Host running even in the absence of any current
20 demand for shipping/mailing address processing. Thus,
there will be no need for start-up when demand arises.
For operating systems that do not provide such a service,
e.g. Windows 95 and 98, a launcher application can be
provided in the host terminal 210 for the same purpose.
25 The launcher application can be included automatically at
the time ABC Host is installed at the terminal 210.

        Installation and other auxiliary software for
Link/Host can be stored at the network server 200 or any
of the client terminals 210, 215, 220, or 225. Instead
30 of at one of the terminals, such as the terminal 210, ABC
Host can be installed at the server 200. Conversely,
while Fig. 4 shows a client-server configuration, ABC
Link/Host can be implemented in the absence of the
network server 200, in a peer-to-peer configuration.

One and the same terminal may include ABC Host
as well as ABC Link, e.g. the remote client terminal 225
in Fig. 4, with a corresponding additional subscription
to ABC Host. In this case, the ABC Link at the terminal

5   225 may use either its own ABC Host or the one provided
via the network

Fig. 5 illustrates shipping/mailing address
validation/standardization in ABC Link/Host prior to use
in labeling. Shown are two network client terminals 215

10 and 220, and three Internet client terminals 225-227, all
in communication with the host terminal 210. From one of
the client terminals, 215, potentially inaccurate or
"dirty" shipping/mailing addresses are assembled in a
marshaling list 500 for checking against AMS data 510

15 from the CD ROM 235. The host 210 returns proposed
"clean" addresses to the marshaling list 500 for
accessing from the client 215.

Without precluding processing of a single
shipping/mailing address individually, the marshaling

20 list 500 facilitates processing of addresses in batches.
This feature can serve to minimize the number of round-
trip communications between the client terminal 215 and
the host 210, thereby enhancing processing efficiency.

Fig. 6 illustrates address standardization

25 processing, either to the successful display of an
address or to failure. From a client terminal 215, a
pre-existing address 601 or a newly entered address 602
can be entered into a marshaling list 603 for submission
605 to the standardizing functionality 606 of the host

30 210. Submission also activates preparation of a license
interface 604. Standardization 606 is contingent on
verifications 607 and 608 that the hardware key 240
remains connected at the host 210 and the requirements of
the license interface are met. If so, the submitted

35 address list is un-marshaled, 609, the submitted

addresses are copied, 610, for AMS processing 611, a
custom-address-marshaling list is prepared for the
standardized addresses and is attached to the submitted
address list, 612 and 613. The resulting list is un-
5   marshaled, 614, for display.

Fig. 7 illustrates address data flow to
printing. Addresses can be created or selected at a
module 701, assembled as Array_Addresses 702, copied as
Array_AddressSearch 703, AMS-processed, 704, e.g. as
10   shown in Fig. 6, and copied as Array_AddressCorrected
705. As AMS-processing may result in several proposed
corrected addresses for one and the same original new
address, display at 706 will prompt the user to select
the one intended, resulting in Array_AddressSelected 707
15   and a key index with respect to Array_AddressCorrected
705. Copying of the finally selected addresses yields
Array_AddressChosen 708 to which business rules can be
applied, e.g. generation of multiples to yield
Array_AddressPrint. Final printing can be subject to
20   printing rules, e.g. how many addresses to print per
sheet of paper in generating labels.

Fig. 8 illustrates address processing for
generating a label. An address 801 can be obtained from
the clipboard 802 where it was placed by a different
25   application 803. An address from the clipboard data can
be parsed, 804, with different parsing rules 805-809
being applied depending on the number of lines of the
address and on the presence/absence of numerals and
special characters, for example. An address 801 can be
30   saved in a database 810, preferably after ABC Host
services 811 have produced the address as standardized,
812. A preferred carrier and class of service, 813, can
be selected for an address 801 or standardized address
812. Printing, 814, can include a 2-dimensional bar code
35   meeting the PDT417 standard, for example.

Fig. 9 illustrates importing of addresses,
activated from a menu 901 and involving browsing, e.g. of
a text file 902, MVP import 903 or database 904. Options
905 include standardizing 906, creating a category 907
5   and including in a database 908. A standarzed address
906 can be selected, 910, for inclusion in the category
907.

Fig. 10 shows workstations 215, 220 and 225
with respective licenses 216, 221 and 226. System
10 communications 1001 result in license registration at the
server application 1002 which includes a license callback
capability for periodic checking on workstations 215, 220
and 225 as to their status under the license. The server
application 1002 can check licenses for functionality
15 1003, and the license can be destroyed, 1004, in case of
lack of authorization. The license is destroyed also in
case a license callback results in failure, in which case
the number of available seats or licenses can be
incremented, 1005, at a dynamic license table 1006. A
20 time license rotator 1007 is in communication with the
dynamic license table 1005, the server application 1002
and the license callback 1002.

Fig. 11 shows the host 210 starting the clock
1101 for periodically changing the license key 1102.
25 Each time a new license key is chosen, the most recent
two keys are saved in a history 1003. Issuance of a new
key initiates callback at the callback queuing table 1104
that is informed by the total number of seats 1105 that
is also referred to by the host 210 in ending an
30 .application if service is requested at too many terminals
as compared with the number of licenses. The host 210
further refers to the authorization number 1106 and
hardware dongle 1107, which both depend on seat options
1008. The ABC license 1009 is established when the
35 application starts. Before an address standardization

1010 can be effected, the key comparison 1011 has to be successful.

Fig. 12 illustrates feature or functionality authorization at the host 210 for a client 215 whose
5   serial number is obtained from a dongle 1201. An authorization code is read, 1202. An encryption engine 1203 is called on for feature decryption, yielding options 1204. The options 1204 are concatenated with internal data 1205 and encrypted to form an encrypted
10 electronic authorization signature 1206. Authorization is established if, at 1207, the authorization code and the encrypted electronic authorization signature are in agreement.

Fig. 13 illustrates processing of a request for
15 address validation and standardization from a terminal 215. Passed in with a standardization request 1301 are a new address list 1302 and the ABC license interface 1303. The ABC host 210 promotes the base interface to the ABC license interface 1304 and ascertains that the request
20 comes with a current authentication "cookie", or at least by one of the most recent two previous cookies. If so, the request for standardizing is acted on, 1306, by actuating AMS 235.

Fig. 14 illustrates license registration for
25 ensuring that the number of client terminals using the system remains limited at all times by the number of licenses. At the terminal 215, the ABC license 1401 is created. Upon connection to the host 210, the license is registered, and a comparison 211 between the total number
30 212 of seats and the available or free number 213 of seats. If no seats are available, 214, the requesting application at the terminal 215 ends. If a seat is available, 215, from callback update table 216 the number of available seats 217 is decremented and a cookie 218 is

issued for later, periodic verification that the terminal
215 continues to be in an authorized state.

Fig. 15 illustrates continuing seat feature
enforcement. Periodically, e.g. every 2 minutes per
5   timer 1501, a new cookie is generated. The current
cookie 1502 is saved, as are the two immediately
preceding values, establishing a cookie history 1503.
Where the callback table 1504 is updated successfully,
the new cookie is forwarded to the corresponding active
10   terminal; otherwise, 1505, the corresponding license is
canceled and the number of available seats is
incremented, 1506.